LOCK STAR – PK

● はじめに 2

● ソフトウェア使用許諾契約書 5

1 LOCK STAR-PK の製品構成 6

2 LOCK STAR-PK の基本仕様 6

3 LOCK STAR-PK の概要 7

4 LOCK STAR-PK の使用手順 9

5 LOCK STAR-PK のデータ構造 10

6 LOCK STAR-PK のデータ編集 10

7 LTPKAdmin でのデータ編集 10

8 LTPKAPI 関数の解説 13

9 LTPEManager による EXE ファイルへのプロテクト 19

10 プログラムの配布について 20

●LOCK STAR-PK 製品トラブルシート 21

管理番号:3.7

● はじめに

『LOCK STARーPK（Protect Key）』は、ロジカルテック株式会社が開発したソフトウェアプロテクト装置です。この装置は、従来のソフトウェアプロテクト機能にハッシュ計算機能をプラスし、より強固なプロテクトを実現した製品です。また、「専用 USB ドライバインストール不要」のハードウェアを採用し、プロテクト実装後のメンテナンスを容易としました。

実際にご使用いただいて、ご意見・ご要望等がございましたら、是非弊社までお寄せ下さい。『LOCK STAR-PK』は、ユーザーの皆様と共に、進化・発展していく所存でございます。

なお、本マニュアルの作成には万全を期しておりますが、万一お気付きの点やご不明な点がございましたら、弊社までご一報いただければ幸いです。

また、マニュアルの内容及び製品のデザインや仕様は、予告なく変更する場合がありますので、予めご了承下さい。

『LOCK STAR』はロジカルテック(株)の登録商標です。

Windows はマイクロソフト社の商標です。
その他、本書に記載されている製品・商品名はそれぞれの発売元あるいは製造元の登録商標です。

本書の版権は、弊社に帰属しますので、御自身でお使いになる以外、許可なく複写・複製等の行為を禁じます。

改訂履歴

作成日	主な改訂内容
2005 年 10 月 04 日	製品マニュアル第 2 版作成。
2006 年 02 月 01 日	「8 LTPKAdmin でのデータ編集」に、注意書きを補足。
2006 年 02 月 15 日	「6 ドライバインストール」に、ドライバ不要 USB キーに関する項目を追加。
2006 年 02 月 24 日	「8 LTPKAdmin でのデータ編集」に、ハッシュ計算機能に関する項目を追加、「9 LTPKAPI 関数の解説」に、ハッシュ計算関数を追加。
2006 年 03 月 15 日	「9 LTPKAPI 関数の解説」に、関数を追加。
2008 年 07 月 23 日	ドライバインストールに関する記述の削除と、それにともなう修正。 USB デザイン変更による修正。
2009 年 01 月 16 日	「8 LTPKAdmin でのデータ編集」と「9 LTPKAPI 関数の解説」に、関数変更に伴い記述を追加。
2009 年 11 月 10 日	Microsoft.NET、Flash 環境で作成された EXE 実行ファイルへのプロテクト対応にともない、「4 LOCK STAR-PK の概要」にユーティリティのファイル構成の追加と「10 LTPEManager による EXE、DLL、OCX ファイルへのプロテクト」の記述の修正。
2010 年 01 月 04 日	専用ドライバが必要なタイプの USB キー利用時の暗号化に関する記述の削除。(「3 LOCK STAR-PK の暗号方式」の削除により、章番号の変更あり。) 64bit の EXE 実行ファイルへのプロテクト対応にともない、「3 LOCK STAR-PK の概要」のユーティリティのファイル構成の変更と「9 LTPEManager による EXE、DLL、OCX ファイルへのプロテクト」の記述の修正。
2010 年 02 月 18 日	「3 LOCK STAR-PK の概要」に LTPEManager.exe の対応 OS の記述の追加。 「4 LOCK STAR-PK の使用手順」と「6 LOCK STAR-PK のデータ編集」にある LTPKAPI.dll のコピー先の記述の修正。
2010 年 08 月 27 日	Mac OS 対応にともない、「3 LOCK STAR-PK の概要」と「製品トラブルシート」に記述を追加。 旧 USB キー用の暗号・復号所要時間測定テストファイル(TestTime.exe)の削除にともない、記述の削除(「10 TestTime でファイルの暗号化、復号化時間を計測する」の削除により、章番号の変更あり。) 「10 プログラムの配布について」(以前と章タイトルを変更)の記述の修正。
2011 年 03 月 29 日	「LTPEManager.exe」のアップグレード(暗号化ロジックに AES を採用/プロテクトファイルの中にアプリケーションの DLL ファイルを挿入可能)にともない、「3 LOCK STAR-PK の概要」と「9 LTPEManager による EXE、DLL ファイルへのプロテクト」(章名変更)と「10 プログラムの配布について」の記述の変更。
2011 年 05 月 23 日	・「LTPEManager.exe」の DLL プロテクト非対応にともない、「9 LTPEManager による EXE ファイルへのプロテクト」に章名を変更。 ・「LTPEManager.exe」の DLL ファイルの挿入について 64bit の EXE ファイルに対応、また「LTPEManager.exe」のオプション圧縮を 3 種類に変更したことにともない、「3 LOCK STAR-PK の概要」と「9 LTPEManager による EXE ファイルへのプロテクト」の記述の変更。 ・「LTPEManager.exe」を使用してプロテクトしたプログラムの配布の際に、64bit のプログラムにおいても、DLL ファイルが不要になったことにともない、「10 プログラムの配布について」の記述の変更。 ・時間制限について、「8 LTPKAPI 関数の解説」に記述の追加。
2011 年 02 月 10 日	・拡張ユーザエリア操作用の関数追加にともない、「7 LTPKAdmin でのデータ編集」の画像の変更、説明の追加。「8 LTPKAPI 関数の解説」に関数説明の追加。 ・USB キーのデザイン変更にともない、「2 LOCK STAR-PK の基本仕様」)の外形寸法と「3 LOCK STAR-PK の概要」の画像と記述の変更。
2012 年 05 月 14 日	・PKAPISHA1HashbyHardEx 関数の追加にともない、「8 LTPKAPI 関数の解説」に関数説明の追加。

2012 年 09 月 03 日	・WPF、Native Code と.NET の Mixed-Mode ファイル対応に伴い、「9 LTPEManager による EXE ファイルへのプロテクト」に説明追加。
2014 年 04 月 02 日	・「3 LOCK STAR-PK の概要」対応 OS に Windows 8/8.1 を追加。 ・「9 LTPEManager による EXE ファイルへのプロテクト」にウイルス防止ソフトについて注意を追加。
2015 年 08 月 07 日	・「3 LOCK STAR-PK の概要」対応 OS に Windows 10 を追加。USB キーの LED ランプに関する注意に記述の追加。

● ソフトウェア使用許諾契約書

この度は、弊社製品をお求めいただき、誠にありがとうございます。ご使用になる前に以下のソフトウェア使用許諾契約書（以下「本契約書」）をよくお読み下さい。

本契約書は、下記に示されたロジカルテック株式会社（以下ロジカルテック）のソフトウェアプログラム（以下「本ソフトウェア」）に関してお客様とロジカルテックとの間に締結される法的な契約書です。本ソフトウェアは、本契約に基づきお客様に提供されるコンピュータソフトウェア及びそれに関連した媒体、ならびに印刷物（マニュアルなどの文書）をいい、「オンライン」または電子文書を含むこともあります。

製 品 名 ： LOCK STAR-PK

第 1 条（使用許諾）

ロジカルテックは、本ソフトウェアをロジカルテックのハードウェア製品「LOCK STAR-PK」のみにおいて使用できる使用権をお客様に許諾します。

第 2 条（無体財産権）

本ソフトウェアに関する著作権その他の知的財産権はロジカルテックに帰属するものであり、本契約は、本ソフトウェアに関する著作権その他の知的財産権をお客様に移転するものではありません。

第 3 条（改変、リバースエンジニアリング、逆コンパイル、逆アセンブルの制限）

お客様は、本ソフトウェアを改変、リバースエンジニアリング、逆コンパイルまたは逆アセンブルしてはなりません。

第 4 条（保証）

ロジカルテックは本ソフトウェアに関して以下の保証のみを行います。なお、この保証は日本国内のみにおいて有効とします。

①お客様が本ソフトウェアをお受け取りになった日から 1 年以内にロジカルテックが本ソフトウェアの誤りの修正を行った時は、ロジカルテックはその修正内容をお客様に提供するものとします。ただし、ロジカルテックがその裁量により情報の提供を決定した場合に限ります。

②本ソフトウェアの記録媒体に、本ソフトウェアの使用に支障をきたすような物理的欠陥があった場合には、お客様が本ソフトウェアをお受け取りになった日から 30 日以内にご連絡いただければ、ロジカルテックは当該記録媒体を無償で交換するものとします。（ただし、ロジカルテックが当該欠陥をロジカルテックの責任と認めた場合に限ります。）これをもって記録媒体に関してロジカルテックが行う唯一の保証とします。

第 5 条（免責）

①ロジカルテックは、本ソフトウェアに関して前条の保証以外にはいかなる保証も行わないものとし、前条の場合以外に、本ソフトウェアに関して発生する問題は、お客様の責任及び費用負担によって処理されるものとします。

②ロジカルテックは、本ソフトウェア及びその関連印刷物並びに本ソフトウェアとともに提供するハードウェアに関して、商業的に有用であること、特定の目的に適合すること等の保証を含み、一切の保証を明示的にも黙示的にも行わないものとします。

③ロジカルテックは、本ソフトウェアの使用により生じる、商業的利益の逸失、商業の支障、その他一切の損害について、いかなる場合でも（たとえかかる損害の生じる可能性があることにつき通知を受けていた場合であっても）いかなる責任をも負わないものとします。

第 6 条（管轄裁判所）

本契約書に関わる紛争は、東京地方裁判所を管轄裁判所として解決するものとします。

1 LOCK STAR-PK の製品構成

LOCK STAR-PK は以下の構成になっております。すべて揃っているかどうかをご確認下さい。万一、不足しているものやマニュアルの落丁などがありましたらロジカルテックまでご連絡下さい。

① ユーティリティソフト

LOCK STAR-PK をお使いいただくためのプログラムがまとめられています。
（1 ユーザー1 枚のみのご提供です。）

② マニュアル（取扱説明書）※PDF ファイル

LOCK STAR-PK の基本的な使用方法や、機能説明、参考となる使い方などについて解説しています。

③ LOCK STAR-PK 本体

企業向けの場合、ユーティリティソフト（マニュアルも含む）は初期販売時のみの添付となります。その後は、LOCK STAR-PK 本体のみの販売となりますのでご了承下さい。

2 LOCK STAR-PK の基本仕様

●温湿度条件

0～70℃、20～80%（ただし、結露しないこと）

●外形寸法

LOCK STAR-PK　：約 W 17mm×L 49mm×H 7mm（キャップ部分含めず）

●重量

LOCK STAR-PK　：約 7g（キャップ部分含めず）

●動作環境

動作電圧　　：4～5.25V
書込回数　　：約 100 万回
データ保存年数　：約 100 年

●ハードウェア保証期間

1 年間（購入日より）

3 LOCK STAR-PK の概要

(1)特長

『LOCK STAR-PK』は、Windows 98/98SE/Me/2000/XP/Vista/7/8/8.1/10 と Mac OS X 10.5 以上が動作するマシンの USB ポートを利用するソフトウェアプロテクターです。

(2)対応 OS

Windows:Windows 98/98SE/ME/2000/XP/Vista/7/8/8.1/10 (USB ポートを標準装備している機種)

※Vista/7/8/8.1/10 は 32bit/64bit 両方に対応

※LTPEManager.exe は、Windows XP 以降の OS に対応

Mac:Mac OS X 10.5 以上 (USB ポートを標準装備している機種)

(3)ユーティリティのファイル構成

LOCK STAR-PK のインターフェースソフト及びユーティリティの構成をツリーで以下に示します。

common ---------- API ファイルフォルダ

- LTPKAPI.dll ------ DLL リンク用ライブラリ
- LTPKAPI.LIB ------ ユーザプログラムのコンパイル/リンク時に使うライブラリファイル
- LTPKAPILib.lib ---- 静的リンク用ライブラリ
- 64bit ---------- 64bit 用の API ファイルフォルダ

exefile ---------- プロテクトツールフォルダ

- DntPackerDll.dll
- LTPackLib.dll
- LtpeApi.dll
- ltpkmpk.dll
- LTPEManager.exe
- LTPK.exe
- x64

Mac OS X ---------- フォルダ

- Demo ---------- Mac 用の Object-C サンプルファイルフォルダ
- Lib ---------- Mac 用のライブラリフォルダ

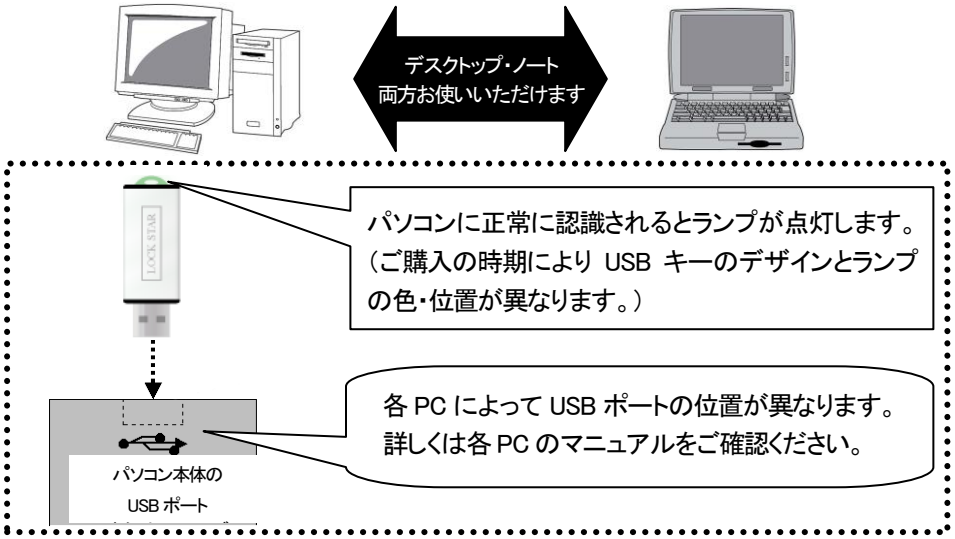
samples ---------- 各種サンプルファイルフォルダ

utility ---------- フォルダ

- LTPKAdmin.exe --- USB キー設定用ツール

(4)『LOCK STAR-PK』接続方法

『LOCK STAR-PK』は、以下の接続図を参考にして接続して下さい。



① 『LOCK STAR-PK』をパソコンに接続すると、新しいデバイスとして自動的に認識されます。

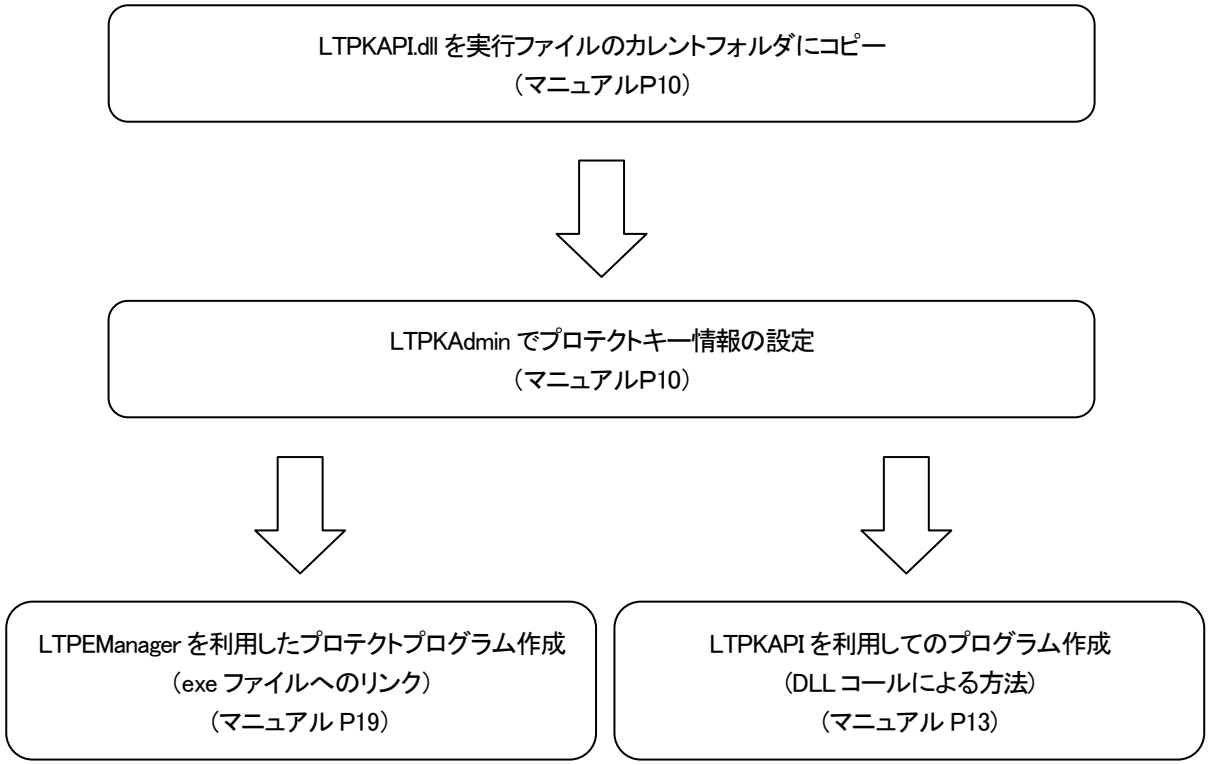
※Windows98/98SE の場合、各 OS のシステム CD-ROM が必要です。

② PC に複数の USB ポートが装備されている場合でも、個々のポートに USB キーを初めて挿入したときに自動的に認識されます。

※初めて USB ポートに挿入した場合、認識されるまでに数秒から数十秒かかります。

※『LOCK STAR-PK』本体を USB ポートに接続しますと、ストラップホールの LED ランプが緑色に点灯します。万が一点灯しない場合は USB ポートの設定の確認や USB キーの抜き挿しを行ってみて下さい。それでも点灯しない場合は、壊れている可能性がありますので利用を停止して下さい。絶対に LED ランプが消灯している状態でご利用にならないで下さい。ただし、OS の仕様によっては、アクセス時のみ点灯する場合があります。

4 LOCK STAR-PK の使用手順



5 LOCK STAR-PK のデータ構造

システムエリア ● USB キー番号 ● 管理者パスワード
コントロールエリア ● 製品番号 ● 読込みフラグ ● 書込みフラグ ● 暗号キー
ユーザーエリア ● データ領域（64 Byte）
拡張ユーザーエリア ● データ領域（7920 Byte） ● Tag 番号領域（8 Byte）

システムエリア（System Area）：USB キー番号と管理者パスワードを管理するエリアです。USB キー番号は出荷時に設定されており、ハードウェアごとに異なります。（ユーザーは変更できません。）管理者パスワードはユーザーが変更できます。

コントロールエリア（Control Area）：製造番号、読込みフラグ、書込みフラグ、暗号キーなどを管理するエリアです。ユーザーが変更できます。

ユーザーエリア（User Area）：ユーザーが自由に使えるエリアです。コントロールエリアの読込みフラグ、書込みフラグによりデータ領域へリード・ライト可能です。

拡張ユーザーエリア（User Area）：ユーザーが自由に使えるエリアです。読み込み可能領域は 7928Byte、書き込み可能領域は 7920Byte です。

6 LOCK STAR-PK のデータ編集

LOCK STAR-PK を使用する前にまず USB キーに製品番号、読込みフラグ、書込みフラグなどを設定しなければなりません。設定には、以下の 2 つの方法があります。

① 付属ユーティリティ内 LTPKAdmin.exe を使用して行う。

② LTPKAPI（関数）を利用してプログラム上より行う。

※LTPKAPI.dll ファイルを実行ファイルのカレントフォルダにコピーしてから編集を開始して下さい。

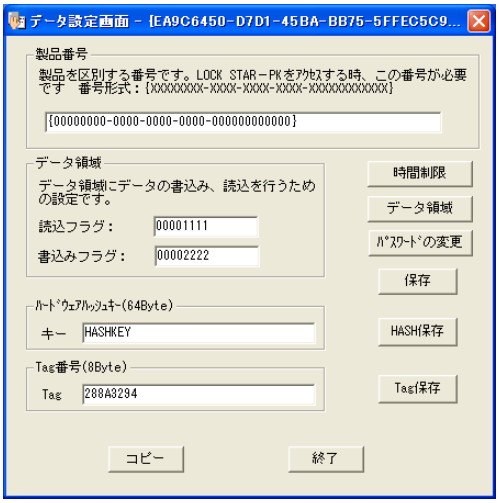
7 LTPKAdmin でのデータ編集

LTPKAdmin は管理者が USB キーのデータを編集するツールです。LOCK STAR-PK を使用する前に USB キーの製品番号、読込み書込みフラグ、暗号キーなどを設定する必要があります。大量設定の為の USB キーのコピーも可能です。

下記手順に沿って、必要項目に設定を行って下さい。

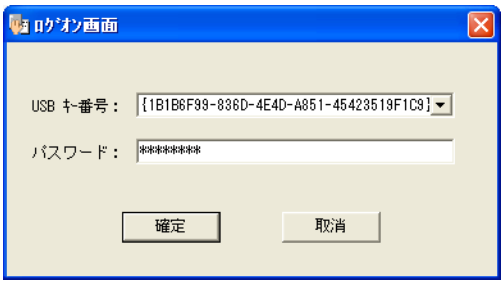
パスワードは、必ず貴社独自の数値に変更して下さい。

（実行には LTPKAPI.dll が必要です。）



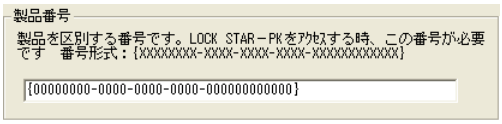
ログオン画面:

USB キーを PC の USB ポートに挿し込み、LTPKAdmin を起動して下さい。管理者パスワードは最大 **15** 桁で初期設定値は **12345678** となっております。複数の LOCK STAR-PK が挿入されている場合は画面右のプルダウンボタンで該当の USB キー番号を選択、パスワードを入力するとデータ設定画面が開きます。



製品番号:

GUID 型。ユーザーが独自に作成できます。
GUIDGEN 等の GUID 作成ツールで作成し、貼り付けて下さい。
（GUIDGEN は Microsoft が提供したツールです。）



時間制限:

USB キーの使用可能期間を設定します。デフォルトは無期限です。

「時間制限の有効」にチェックし、開始、終了期間を画面右のプルダウンボタンでそれぞれ指定し、【確定】ボタンをクリックすると設定内容を確定し画面を閉じます。【保存】ボタンをクリックし確定した使用期限をキーに書き込んで下さい。

この時間制限は API 中の SKCONTROLDATA 構造体によりチェック可能です。また、下記の方法で得られた dwTime と比較してください。

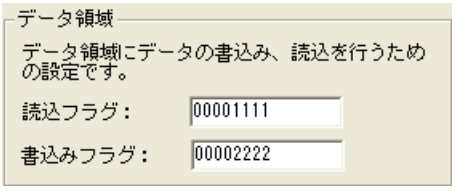
```
SYSTEMTIME  st;
GetLocalTime(&st);
DWORD dwTime = MAKELONG(MAKEWORD(st.wDay, st.wMonth), st.wYear);
```

データ領域:

読込みフラグ：ユーザーエリアの読込み用フラグ（最大 8 桁 HEX）
書込みフラグ：ユーザーエリアの書込み用フラグ（最大 8 桁 HEX）

それぞれ、任意の数値を設定し、【保存】ボタンをクリックしキーに書き込んで下さい。

※読込み、書込みフラグは、必ず異なる数値にて設定して下さい。



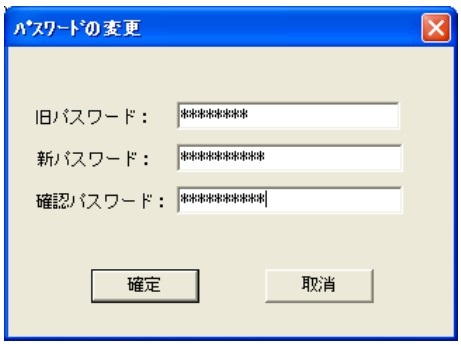
データ領域:

ユーザーエリア内のデータ領域に最大 64 バイトを任意に設定することができます。【データ領域】ボタンをクリックすると「データ領域」画面が開きます。任意の数値を設定し、【確定】ボタンをクリックすると設定するデータを確定し画面を閉じます。【保存】ボタンをクリックし確定したデータをキーに書き込んで下さい。

パスワードの変更:

管理者パスワードの変更をします。最大 15 桁を設定できます。初期設定値は 12345678 となっております。

現在のパスワードを入力し、新しいパスワードを確認の為 2 回入力し、【確定】ボタンをクリックします。「パスワードを変更しました」のメッセージが表示されれば、変更は完了です。



保存:

設定又は、変更されたデータを USB キーに保存します。

「Write to USB key OK」のメッセージが表示されれば、保存は完了です。

コピー:

USB キー内データをコピーします。

設定済みの USB キーを接続し、LTPKAdmin を起動して「データ設定画面」を開きます。コピー元の USB キーを抜き、コピー先の新しい USB キーを挿入し【コピー】ボタンをクリックします。

「Copy to USBkey OK,Count=コピー数」が表示されるとコピーが完了です。複数の USB キーにコピーを行う場合、続けてキーを挿入し操作を繰り返して下さい。

※コピーを行う場合、コピー元とコピー先のキーに、必ず同じパスワードを設定して下さい。パスワードが異なる場合コピーに失敗します。

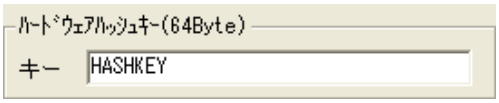
HASH 保存:

「ハードウェアハッシュキー」欄に、任意の英数字を入力し【HASH 保存】ボタンで、書き込みます。

「Write key success」が表示されれば、書き込み完了です。

キーは最大 64 バイトを設定可能です。

※設定した HASHKEY を設定画面で確認することはできません。



Tag 保存:

「Tag 番号」欄に、外装に刻印されている 8 桁の Tag 番号を入力し、【Tag 保存】ボタンで書き込みます。

「Write Tag success」が表示されれば、書き込み完了です。

最大 8 バイトを設定可能です。

キーの外装に刻印されているタグ番号を管理できます。

終了:

LTPKAdmin を終了します。

8 LTPKAPI 関数の解説

※LTPKAPI.DLL 関数の使用については、Samples フォルダの中のデモファイルをご参照下さい。

BOOL WINAPI SKAPIInitialize();

初期化関数、LTPKAPI.DLL を利用する場合、必ず最初に初期化して下さい。
Return:
TRUE: 成功
FALSE: エラー

void WINAPI SKAPIClose();

LTPKAPI のクローズ関数。LTPKAPI を終了する時、コールして下さい。

BOOL WINAPI SKAPIChangeAdminPassword(const GUID* pidDevice, const BYTE* pbyOld, const BYTE* pbyNew);

管理者パスワードを変更する関数です。
pidDevice: USB キー番号 GUID 型
pbyOld: 旧パスワード
pbyNew: 新しいパスワードをセットします。NULL の場合、SKAPIChangeAdminPassword は pbyOld のパスワードについてチェックを行うだけで、書換えは行ないません。
Return:
TRUE: 成功
FALSE: エラー

BOOL WINAPI SKAPIGetControlData(const GUID* pidDevice, const BYTE* pbyPassword, SKCONTROLDATA* pcd);

コントロールエリアデータの読込み関数です。
pidDevice: USB キー番号 GUID 型
pbyPassword: 管理者パスワード(パスワードが NULL の時、pidDevice と pcd 構造体の dwStartDate、dwEndDate のみを返します)
pcd: コントロールエリアデータ、成功した場合、pcd にデータがセットされます。
SKCONTROLDATA メンバー:
guidDevelop: GUID, 製品番号 GUID
dwReadPassword: DWORD、読込みフラグ
dwWritePassword: DWORD、書込みフラグ
byEncryptKey: BYTE[16], 暗号キー
dwValidateOption: 時間制限の有効フラグ(1 の時有効、それ以外は無効)
dwStartDate: 開始期間
dwEndDate: 終了期間
Return:
TRUE: 成功
FALSE: エラー

BOOL WINAPI SKAPISetControlData(const GUID* pidDevice, const BYTE* pbyPassword, const SKCONTROLDATA* pcd);

コントロールエリアデータの書込み関数です。

pidDevice: USB キー番号 GUID 型

pbyPassword: 管理者パスワード

pcd: コントロールエリアデータ

SKCONTROLDATA メンバー:

guidDevelop: GUID, 製品番号 GUID

dwReadPassword: DWORD、読込みフラグ

dwWritePassword: DWORD、書き込みフラグ

byEncryptKey: BYTE[16], 暗号キー

dwValidateOption: 時間制限の有効フラグ(1 の時有効、それ以外は無効)

dwStartDate: 開始期間

dwEndDate: 終了期間

Return:

TRUE: 成功

FALSE: エラー

BOOL WINAPI SKAPIEnumDevices(ENUMDEVICEPROC lpEnumProc, DWORD dwParam);

USB キーの列挙関数です。

DwParam: ユーザー定義パラメータ

LpEnumProc: 呼出関数のポインター, SKAPIEnumDevices が USB キーを見つけた時、この関数を呼び出します。

BOOL WINAPI EnumProc(GUID* pidDevice, DWORD dwParam);

PidDevice: USB キー番号 GUID 型

DwParam: SKAPIEnumDevices の dwParam

Return:

TRUE: SKAPI が継続列挙

FALSE: 列挙停止

Return

TRUE: 成功

FALSE: エラー

BOOL WINAPI SKAPIGetUserData(const GUID* pidDevelop, const DWORD dwReadPassword, BYTE* pbyData);

ユーザーエリアのデータ読込み、および使用制限時間のチェック。(一度期限切れとなったキーについて LTPKAdmin もしくは SKAPISetControlData 関数を利用し、期間の再設定を行ってください。)

pidDevelop: 製品番号 GUID 型

dwReadPassword: 読込みフラグ

pbyData: データエリアポインタ(64Byte)

Return

TRUE: 成功

FALSE: エラー

BOOL WINAPI SKAPISetUserData(const GUID* pidDevelop, const DWORD dwWritePassword, const BYTE* pbyData);

ユーザーエリアのデータ書込み関数です。

pidDevelop: 製品番号 GUID 型

dwWritePassword: 書込みフラグ

pbyData: データエリアポインタ(64Byte)

Return

TRUE: 成功

FALSE: エラー

BOOL WINAPI PKAPIGetUserData(const GUID* pidDevice, const GUID* pidDevelop, const DWORD dwReadPassword, BYTE* pbyData);

同一製品番号のキーを挿している時、指定したキー番号と製品番号によりデータエリアからデータの取得、および使用制限時間のチェック(一度期限切れとなったキーについて LTPKAdmin もしくは SKAPISetControlData 関数を利用し、期間の再設定を行ってください。)

pidDevice: USB キー番号 GUID 型

pidDevelop: 製品番号 GUID

dwReadPassword: 読込みフラグ

pbyData: データエリアポインタ(64Byte)

Return

TRUE: 成功

FALSE: 失敗

BOOL WINAPI PKAPISetUserData(const GUID* pidDevice, const GUID* pidDevelop, const DWORD dwWritePassword, const BYTE* pbyData);

同一製品番号のキーを挿している時、指定したキー番号と製品番号によりデータエリアにデータの書込み

pidDevice: USB キー番号 GUID 型

pidDevelop: 製品番号 GUID

dwWritePassword: 書込みフラグ

pbyData: データエリアポインタ(64Byte)

Return

TRUE: 成功

FALSE: 失敗

BOOL WINAPI PKAPIGetUserDataEx(const GUID* pidDevice, const GUID* pidDevelop, const DWORD dwReadPassword, DWORD dwOffset, BYTE* pbyData, DWORD dwLength);

拡張ユーザエリアからデータの取得、および使用制限時間のチェック(一度期限切れとなったキーについて LTPKAdmin もしくは SKAPISetControlData 関数を利用し、期間の再設定を行ってください。)

pidDevice: USB キー番号 GUID 型

pidDevelop: 製品番号 GUID

dwReadPassword: 読込みフラグ

dwOffset: オフセット(オフセットは 0 から 7927 まで)

pbyData: データエリアポインタ

dwLength: 読み込みサイズ (オフセット+読み込みサイズは 7928 以下)

Return

TRUE: 成功

FALSE: 失敗

BOOL WINAPI PKAPISetUserDataEx(const GUID* pidDevice, const GUID* pidDevelop, const DWORD dwWritePassword, DWORD dwOffset, const BYTE* pbyData, DWORD dwLength);

拡張ユーザエリアへのデータの書き込み、および使用制限時間のチェック(一度期限切れとなったキーについて LTPKAdmin もしくは SKAPISetControlData 関数を利用し、期間の再設定を行ってください。)

pidDevice: USB キー番号 GUID 型

pidDevelop: 製品番号 GUID

dwWritePassword: 書込みフラグ

dwOffset: オフセット(オフセットは 0 から 7919 まで)

pbyData: データエリアポインタ

dwLength: 書き込みサイズ (オフセット+書き込みサイズは 7920 以下)

Return

TRUE: 成功

FALSE: 失敗

以下は、ドライバ不要 USB キーに対応するハッシュ計算関数です。同一ハッシュ暗号キーで同じデータに対してハッシュ計算を行う場合、ソフトウェアハッシュ計算とハードウェアハッシュ計算の結果は同じです。

BOOL WINAPI PKAPIWriteKey(const GUID* pidDevice,LPCSTR lpAdminPassword, BYTE* pucKey,DWORD ulKeySize);

ハードウェアハッシュ計算用のハッシュ暗号キーの設定

pidDevelop: 製品番号 GUID

pbyPassword: 管理者パスワード

pucKey: ハッシュ暗号キーのバッファポインター

ulKeySize: ハッシュ暗号キーサイズ

Return

TRUE: 成功

FALSE: 失敗

BOOL WINAPI PKAPISHA1HashbyHard(const GUID* pidDevelop, DWORD dwReadPassword, BYTE* pucData,DWORD ulLength,BYTE* pucDigest);

設定したハードウェアハッシュ暗号キーに用いて、ハードウェアによりデータのハッシュ計算を行った結果を返す。

pidDevelop: 製品番号 GUID

dwReadPassword: 読込みフラグ

pucData: ハッシュ計算しようとするデータバッファポインター

ulLength: データバッファサイズ

pucDigest: 20Byte のハッシュ計算結果を返すバッファポインター、失敗した場合は NULL をセットされる。

Return

TRUE: 成功

FALSE: 失敗

BOOL WINAPI PKAPISHA1HashbyHardEx(const GUID* pidDevice, const GUID* pidDevelop, DWORD dwReadPassword, BYTE* pucData,DWORD ulLength,BYTE* pucDigest);

設定したハードウェアハッシュ暗号キーに用いて、ハードウェアによりデータのハッシュ計算を行った結果を返す。

pidDevelop: 製品番号 GUID

dwReadPassword: 読込みフラグ

pucData: ハッシュ計算しようとするデータバッファポインター

ulLength: データバッファサイズ

pucDigest: 20Byte のハッシュ計算結果を返すバッファポインター、失敗した場合は NULL をセットされる。

Return

TRUE: 成功

FALSE: 失敗

BOOL WINAPI PKAPISHA1HashbySoft(const GUID* pidDevelop,DWORD dwReadPassword,BYTE* pucData,DWORD ulLength,BYTE* pucKey,DWORD ulKeyLength, BYTE* pucDigest);

渡してきたデータをソフトウェアハッシュ暗号計算によりハッシュ計算を行う

pidDevice: USB キー番号 GUID 型

pidDevelop: 製品番号 GUID

dwReadPassword: 読込みフラグ

pucData: ハッシュ計算しようとするデータバッファポインター

ulLength: データバッファサイズ

pucKey: ハッシュ暗号キーのポインター

pucDigest: 20Byte のハッシュ計算結果を返すバッファポインター、失敗した場合は NULL をセットされる。

Return

TRUE: 成功

FALSE: 失敗

9 LTPEManager による EXE ファイルへのプロテクト

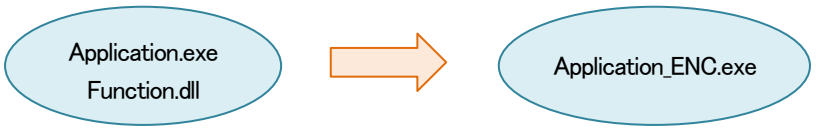
LTPEManager は実行ファイルに直接にプロテクトプログラムを挿入するプロテクトツールです。
本ツールを利用することにより、実行ファイルのヘッダーを外されても、挿入された DLL を取り出されなければ実行できません。
さらに AES 暗号ロジックを採用しているため、より強力なセキュリティを実現しています。
また、API プロテクトと LTPEManager.exe を合わせて使用し、2 重のプロテクトをかけることも可能です。
LOCK STAR-PK キーを挿入し、CD-ROM 内「exefile」フォルダから、LTPEManager.exe を実行して下さい。

LTPEManager の画面

①元のファイル名:
プロテクトをかける元のファイル名を入力します。(右のボタンで参照、または直接入力して下さい。)

②新しいファイル名:
プロテクトをかけ新規作成するファイル名を入力します。(右のボタンで参照、または直接入力して下さい。ファイル名を変更しない場合、元のファイル名に_ENC が追加されます。)

③挿入 DLL ファイル:
プロテクトをかける実行ファイルに挿入したい静的リンクで呼び出される DLL ファイル名(14Byte 以内)を入力します。(右のボタンで参照、または直接入力して下さい。)
実行ファイルに DLL ファイルを挿入し、合わせてプロテクトを行うことで、セキュリティ強度を高める事ができます。
DLL ファイルの挿入イメージは下図の通りです。
ただし、DLL によっては、対応できないものもありますので必ず動作を確認して下さい。



元のアプリケーション　　プロテクトされたアプリケーション

④製品番号:
LTPKAdmin.exe によりキーに設定された製品番号(GUID 型)を入力してください。(マニュアル 11 ページをご参照下さい。)

⑤読込みフラグ:
事前に設定したユーザーエリア内データ領域の読込み用フラグ(最大 8 桁 HEX)を入力して下さい

⑥チェック時間:
プロテクトしたファイルの実行中に USB キーの有無をチェックする時間を設定します。
0 から最大 30 分まで設定可能で、チェック時に USB キーが挿入されていない場合エラーメッセージが表示され、3 分後にプログラムは終了します。終了時にはメッセージは表示されませんので、ご注意下さい。
チェック時間を 0 分に設定した場合は、起動時のみ USB キーの有無をチェックします。

⑦オプション:
セキュリティの設定です。

「全コード暗号化」にチェックすると、プロテクト実装後のコードが自動的に暗号化されます。
「全コード圧縮」にチェックすると、3 種類の状態を選択できます。
■の場合:リソース以外を圧縮する。
□の場合:すべて圧縮しない。
☑の場合:全コードを圧縮する。
「セクション再配置」「リロケーション不能」にチェックすることで、セキュリティ強度を上げることができます。
ただし、プログラムによって使用できない場合がありますので、プロテクトをかけることができない場合は、調整してください。
「.NET Mode (WPF)」は、WPF を使用したプログラムにプロテクトする際にチェックしてください。
「実行時のエラーメッセージの表示」にチェックし、メッセージを入力することで、実行時のキーチェックエラー時に発生するエラーの内容を設定できます。

⑧開始:

全ての設定が終了したら【開始】ボタンをクリックして下さい。メニュー画面下部ウィンドウに処理工程が表示され、最後に「success!」と表示されたらプロテクトが完了し、プロテクト済みの新しいファイルが作成されます。ファイルのパスを指定しない場合、LTPEManager と同一ディレクトリに新しいファイルが作成されます。
「Hardware not found!」が表示される場合は、入力項目の誤りや、LOCK STAR-PK が挿入されていない事などが考えられます。入力項目の再確認、LOCK STAR-PK の挿入を確認して下さい。
最後に「success!」が表示されない場合は、表示されるメッセージを巻末のトラブルシートに明記の上、弊社製品サポート部までお問い合わせ下さい。
また、.NET Framework の Mixed-Mode ファイルにプロテクトをかける場合、「Enc Mixed .NET PE Successfully. Put "ltpkmpk.dll" together with it!」が表示されます。このメッセージが表示された場合、プロテクト済みのファイルと同じ場所に作成される ltpkmpk.dll を必ず一緒に配付してください。

⑨About:

本ツールの製品バージョンを表示します。

⑩Exit:

本ツールを終了します。

⑪プロテクト実装ファイルの動作確認:

LTPEManager によるプロテクトが成功すると、USB キーを接続していない場合、アプリケーションは起動しません。

注意事項

- **.Net Framework を使用したプログラムにプロテクトをかける場合は、プログラムの実行環境に.Net Framework 4 以上をインストールする必要があります。**
- **プロテクトをかけたアプリケーションを起動中に各種デバックツールを起動させた場合、アプリケーションが強制終了してしまう場合があります。**
- **プログラムによっては実行ファイルに対応できない場合がありますので、ご了承下さい。**
- **プロテクトの仕組み上、一部のウイルス防止ソフトに誤検知される場合があります。除外設定等、ご対応下さい。**

10 プログラムの配布について

プロテクトしたプログラム(EXE)を配布する際に必要な DLL は次の通りです。

プロテクト・プログラム	LTPEManager を使用した場合	LTPEManager を使用しない場合
32bit の場合	無し	LTPKAPI.dll
64bit の場合	無し	LTPKAPI.dll

その際、LTPKAPI.dll を実行ファイルのカレントフォルダにインストールすることをお勧めします。(ただし LTPKIAPILib.lib を使用する場合は LTPKAPI.dll は不要です。)

●LOCK STAR-PK 製品トラブルシート

整理番号 PK-
※ロジカルテック使用欄

何かトラブル等がございましたら、このトラブルシートにご記入後、弊社まで FAX または E-mail にてお送り下さい。折り返し担当者よりご連絡させて頂きます。

ロジカルテック株式会社　製品サポート部　行

年　月　日
FAX:03－5496－0213
E-mail:support@logicaltech.co.jp

御 社 名:　　部 署 名:

御担当者名:　　E-Mail:　　@

T E L:　　F A X:

■商品名　LOCK STAR-PK

■ユーティリティのバージョン:Ver.

■OS
☐ Windows 98　☐ Windows 98SE　☐ Windows Me
☐ Windows 2000 Pro　☐ Windows XP Home　☐ Windows XP Pro(サービスパック　)
☐ Windows Vista(エディション　サービスパック　)
☐ Windows 7(エディション　サービスパック　)
☐ Windows 8(エディション　サービスパック　)
☐ Windows 8.1(エディション　サービスパック　)
☐ Windows 10(エディション　サービスパック　)
☐ Mac OS X(バージョン　)

■トラブルの現象: